

Panel:

Programming in the Small or What?

Dennis Brylow (Marquette)

Tony Hosking (Purdue)

Lenny Pitt (UIUC)

Bruce Weide (Ohio State)

Traditional programming courses

- * Step-by-step acquisition of basic building blocks:

- * variables

- * expressions

- * assignment

- * subroutines

Traditional programming courses

- * Emphasis on:

- * Control structures vs abstraction

- * Simple explicit data structures: arrays, lists, etc.

- * Subroutines as unit of abstraction

And yet...

- * Modern languages like Java, C#, Python support rich built-in abstractions
- * Heavily reused by real programs
- * Tension: teaching compositional programming versus bottom-up

Even more...

- * Programming in the small emphasizes implementation over design
- * Should we also be teaching scientists to think computationally by composing computational abstractions?
- * Real programs "maintained" not "written"

And more...

- * What about programming in the medium and large?
- * Extreme programming
- * Testing (eg, unit tests)
- * UML
- * Software process

Computational thinking?

- * More than programming...
- * Communication of computational thinking in the language of software designers?
- * What else should it be... ?